Evaluation of Spatial Sound Description in Virtual Environments

David Murphy
Computer Science Department
University College, Cork
d.murphy@cs.ucc.ie
http://www.cs.ucc.ie

20th September, 1999

ABSTRACT

Recent advances in Spatial Sound have made important contributions to Virtual Environment research. The evolution of spatial sound description from VRML to MPEG-4 is examined and some of the shortcomings of these languages are highlighted. An auditory framework for examining the process of sound spatialization is introduced and existing spatial description languages are measured against this. Different models of spatial sound representation are reviewed and questions are raised about certain spatial sound attributes. The merits of using a subjective lexicon to describe spatial characteristics are also examined.

0 INTRODUCTION

Research into spatial sound and its application in Virtual Reality has been active for a number of years. Historically, several projects, including the DIVA project (Helsinki), the Spatialisateur project (IRCAM), and the DIVE Auralizer (SICS) have made great improvements in the different areas of spatial sound description and presentation. However, most research has been limited to proprietary systems. Advancements made in spatial sound research have been incorporated into two ISO/IEC Multimedia standards, VRML97 [1] and MPEG-4 [2]. Both of these standards are based upon hierarchical languages designed to describe virtual worlds that are platform-independent.

Both VRML97 (Virtual Reality Modeling Language 1997 standard) and MPEG-4 (Motion Picture Experts Group, standard 4) control the description of the scene by prescribing the decoding process of the scene information. However the encoding is not specified by either language, allowing the developer the freedom to use whatever implementation is more appropriate, or more efficient, at the time of development. Hence, these two scene description languages are unconcerned with the underlying implementation method.

1.1 OBJECT ORIENTED APPROACH

Each item in a virtual environment can be described as an object or Node. This approach allows for easy manipulation and duplication of items. It is possible then to have multiple instances of the same object. In MPEG-4 these objects, when grouped together, are collectively called Audio-

Visual Objects. A complex object that contains sub-objects is termed a Compound Object. A Sound Object, for example, could be declared with a particular set of characteristics (these are defined in sub-nodes) that could then be reused in different situations. So it is possible to create a default sound object with a particular set of spatial parameters that could be reused in various contexts.

1.2 LOW-LEVEL APIs

Many manufacturers have developed low-level APIs (Application Programmer Interface) which have spatial sound rendering functionality. Among the more popular are DirectX (Microsoft), Java3D (Sun) and Aureal's A3D. In general, these APIs are finely tuned and sophisticated, however, they are proprietary and, in most cases, machine dependent. The focus of this paper is on specifications that have been standardized and are platform independent. Therefore further discussion of APIs is not relevant.

2 SOUND MODELS

The VRML and MPEG-4 (v.1) sound presentation model is based upon the environment and the listening position. Dependency upon physical properties is probably the main characteristic of a Virtual Reality sound model.

In a virtual environment the scene, or world, is normally dependent upon environment modeling. A virtual room, for example, could be composed of four walls, a ceiling and a floor. Associated with this room are the room's acoustic properties i.e. reverberation, absorption, etc. Hence, sound objects are normally correlated with some visual/physical cue. This model has been described as 'acoustic environment modeling' [3]. In recent times acoustic environment modeling has been complemented by a 'perceptual approach' [4]. The perceptual approach allows for absolute sound to be rendered irrespective of the environment; for example, one can render a large-reverberant sound in a small confined space.

When describing sound in an artificial environment one tends to use familiar, everyday terms to represent the sensation. However, artificial environments are not necessarily comprised of the familiar, indeed recent trends in Virtual Reality show that the worlds being created do not seek to imitate real environments, instead they are creating new synthetic environments constrained only by resources and imagination [5].

Traditionally we rely upon environment and visual cues (as described above) to determine the spatial characteristics of a sound. This approach, while useful, is limited when one is faced with a new and unfamiliar environment. Consider the user, or avatar, that is taking part in a multi-user virtual environment, not all of the objects, including sounds, are rendered from his or her perspective.

Just as sound enhanced the experience of watching movies, spatial sound increases the sensation of realism in a virtual environment. If a virtual environment merely contained visual objects and scene geometry it would come across as very bland and fall short in any attempt to immerse the user completely. Even stereo sound would fail to create an acceptable level of realism. The user needs to be enveloped by sound in-order for a convincing degree of immersiveness to be attained. Hence the importance of spatial sound within VR.

2.1 Auditory Framework

In order to manipulate the presentation of sound in either the ‘acoustic environment model’ or the ‘perceptual model’ we need to deconstruct the auditory process into stages with distinct parameters/properties. One possible approach is to divide the process into four discrete stages: Source, Medium, Environment, and Listener. Table 1 contains a simple listing of some of the possible properties for each of the defined stages. Whichever sound model is used it will almost certainly be composed of some combination of the four stages.

When describing the properties of a sound source we describe attributes such as the sound’s location, intensity and radiation pattern. But there are other, equally important attributes that need to be defined in-order to create a cogent spatial experience; these include the environment’s acoustic fingerprint. These environment cues (attenuation, reverberation, absorption, etc) are used in conventional room-simulation (auralization).

Table 1

Source	Medium	Environment	Listener
Location	Velocity	Reverberation	Shadowing
Directivity	Absorption	Reflection	Filtering
Intensity	Filtering	Occlusion	Cognitive - Process
			Visual - Association

3 VRML

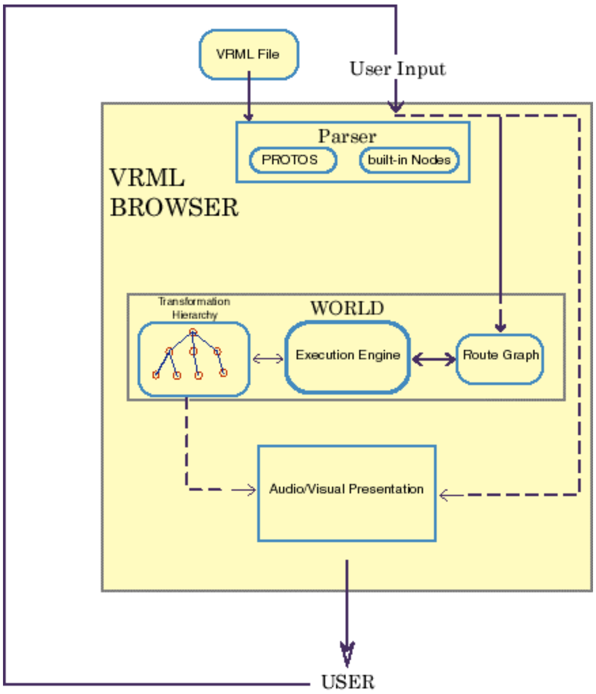
VRML originated from a 3D graphics format, known as Open Inventor (OI), developed by Silicon Graphics, Inc (SGI). It was published as an ISO standard in May 1995. The first version of the specification concentrated on visual rendering and support for sound was not included.

VRML was initially designed as a Markup Language similar to that of SGML and HTML. However, it became apparent early on that, due to its reliance upon geometric models, it was more akin to a Modeling Language than a Markup Language. Consequently the name was changed. VRML was designed to be retrieved from a server and read by a local Browser, much the same as HTML.

A VRML world could be described as a scene composed of interactive dynamic media objects. The structure of a VRML scene is described from the top down (tree structure) but is rendered from the bottom up. At the lowest level of the tree, each of the different media objects is exclusively declared. These objects are then grouped together according to their relationships within the scene. Objects, or Nodes as they are called in VRML, are composed of fields, which are the various parameters of a Node. A Node can also consist of sub-nodes (children-nodes) and, when grouped together, these create a local coordinate system. The scene description file is passed to a Browser; this parses the scene information from the file and renders the virtual world as shown in Figure 1.

A point to note about this process of parsing is that each time a scene is changed or updated the information is re-read from the file again; this introduces latency issues, especially when rendering complex shared scenes over networks.

Figure 1

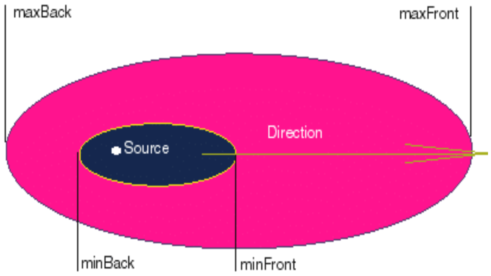


Conceptual Model of a VRML Browser (from VRML97 spec)

VRML has undergone a number of changes since its inception. Since 1995, the introduction of sound has been slow and evolutionary. The first sound-aware version of VRML was version 2 (1996). Sound manipulation was primitive and did not facilitate spatial rendering. It was not until VRML97 (1997) that sound functionality was advanced to a stage where spatial attributes could be defined in terms of a 3D coordinate system.

The **Sound** Node in VRML97 contains ten fields (see Appendix A). Four of these ('minFront', 'minBack', 'maxFront' and 'maxBack') define the radiation pattern of a sound object. This pattern is restricted to an elliptical shape and, consequently, sound objects are treated as directional sounds [7]. The direction of the sound, which corresponds to the apex of the ellipsoid (see Figure 2), specifics the path along which the direct sound will travel. This vector is specified in the 'direction' field.

Figure 2



Elliptical Model for Sound Source

The sound source is bounded by two ellipsoids within which a linear attenuation is performed. The inner ellipsoid has an intensity of 1.0 (the maximum level, 0dB) and the minimum level of the sound 0.0 (-20dB) is determined by the outer ellipsoid. However, the cut-off at –20dB is abrupt and very audible. Briefly, the attenuation can be calculated using the following formula:

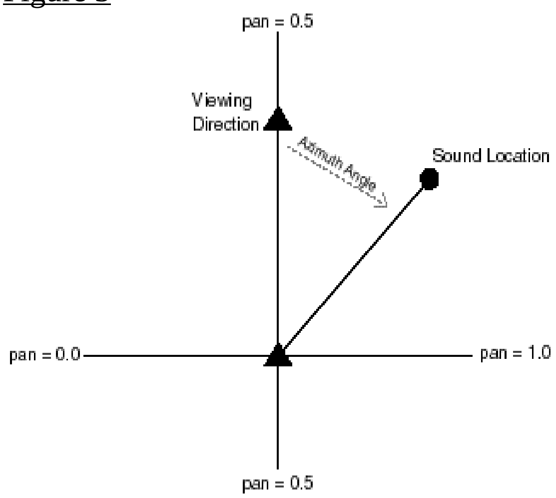
$$\text{Attenuation} = -20 \times (d^{1} / d^{11}) \qquad [1]$$

where d^1 is the distance along the direction vector, measured from the inner ellipsoid boundary to the viewer, and d^{11} is the distance along the direction vector measured from the inner ellipsoid boundary to the outer ellipsoid boundary. The outer ellipsoid also behaves as an acoustic proximity sensor and when the user traverses this perimeter the sound is activated.

The location of the sound source is specified in the 'location' field; this is a 3D coordinate position given in the X, Y and Z-axis. The final attribute, which activates the spatialization of the sound object, is the 'spatial' field. The 'spatial' field is a Boolean field and if set to TRUE will activate the terminal system's spatial mechanism. In relation to the spatial sound capability of the end-user's terminal, the VRML97 standard specifies that as a minimum "Browsers shall at least support stereo panning of non-MIDI sounds based on the angle between the viewer and the source" [1]. For simple stereo panning, the location of the source is mapped to the XZ planes to determine the azimuth of the source in relation to the viewer (Figure 3). This angle is then assigned a pan value between 0.0 and 1.0. However, it is recommended that the Browser should use a more sophisticated technique of spatialization than basic amplitude panning [1].

One of the main shortcomings of sound spatialization in VRML is that there is no explicit use of height/elevation information or even recommendations for its use. Height information could be derived from a comparison of the listener position and the sound source location on the Y-axis. This information could then be passed to the Browser's rendering engine.

Figure 3



VRML Stereo Panning
from VRML97

In terms of the four stages of the auditory process, VRML only address the first stage while the others are neglected. It is clear from this that, even at this stage of its development, VRML has a very basic sound spatialization functionality.

4 MPEG-4

MPEG (Motion Picture Experts Group) is a working group of an ISO/IEC subcommittee that generates generic standards for multimedia. In particular, MPEG defines the syntax of low bitrate video and audio bit streams, and the operation of codecs. MPEG has been working for a number of years on the design of a complete multimedia toolkit, which can generate platform independent, dynamic interactive media representations. The first stage of this development was in December 1998 with the release of version 1 of MPEG-4.

In the MPEG-4 standard, the various media are encoded separately; this allows for better compression, the inclusion of behavioral characteristics and also enables user-level interaction. Instead of creating a new scene description language the MPEG organization decided to incorporate VRML. The presentation of the scene and associated interactions are controlled by MPEG-4's binary language, BIFS (Binary Format for Scenes).

Within MPEG-4, each media type has its own stream of information and its rendering parameters are contained in the BIFS stream. Accordingly, sound is encoded in one elementary stream while its spatialization properties are encoded in the BIFS stream. This separation allows for varying levels of QoS (Quality of Service) and presentation Profiles. For example, if the user's machine does not support advanced audio spatialization then MPEG allows for a graceful degradation in the spatial attributes of the audio presentation.

As mentioned above, MPEG-4 is a binary language, hence it does not have to contend with the latency issues associated with VRML. This allows for near real-time rendering, instantaneous response to user interaction and the streaming of dynamic media. Some of the applications suggested for MPEG-4 include multi-user conferencing, which could render the spatial position of each participant's voice, and virtual tours, i.e. of famous buildings, with faithful representations of the original's acoustic signature.

VRML's scene description capabilities were not very sophisticated so MPEG extended the functionality of the existing VRML nodes and incorporated new nodes with advanced features. Support for advanced sound within the scene graph was one of the areas developed further by MPEG.

4.1 MPEG-4 version 1

The **Sound** Node of MPEG-4 is quite similar to that of the VRML **Sound** Node. The main difference being in the 'source' field. In MPEG-4 this field can contain any, or all, of the AudioBIFS Nodes as the input.

The addition of AudioBIFS Nodes has increased the capability of sound spatialization within the scene. The new Audio Nodes are **AudioSource**, **AudioMix**, **AudioSwitch**, **AudioDelay**, **AudioFX** and **AudioClip** (see Appendix B). Examination of these Nodes show that version 1 of MPEG-4 is based on the physical sound model described earlier. In terms of the four stages of the auditory process (Table 1) version 1 performs much the same as VRML as it only addresses the first of the four stages of the auditory process.

The inclusion of the 'phaseGroup' field into the AudioBIFS Nodes facilitates the rendering of multi-channel sound files; this is achieved by preserving the correct phase relationships of the audio channels. In VRML the listener's position was assumed to be the same as the viewing position. In MPEG-4 a new **ListeningPoint** Node was added so that the listening position need not be the same as the viewing position. The spatial locations of sound sources are calculated relative to the **ListeningPoint**.

The **AudioFX** node generates signal processing effects via Structured Audio (a DSP language which is a derivative of CSound and a subsection of MPEG-4 Audio). These custom effects can be applied to any of the sound input channels. Some of the effects possible, for example, are filters, artificial reverberation, delay, equalization, etc. Structured Audio is extensible and hence a developer can include arbitrary effects irrespective of the underlying hardware. The remaining AudioBIFS Nodes deal primarily with the grouping and mixing of the audio sources.

Although MPEG-4 version 1 is undoubtedly an improvement over VRML, in terms of sound spatialization there are a few caveats:

- The 'shape' of the radiation pattern for a sound source is still restricted to an elliptical form.
- As sound channels can be phase-grouped anything other than scaling of the sound would have a disastrous affect upon the spatialization of the sound. For example, further filtering or repeated spatialization

would destroy the phase relationships between related channels.

- Likewise, over use of effects in Structured Audio would presumably blur the localization of the sound source and make nonsense of its spatialization attributes.

4.2 MPEG-4 version 2

The evolution of sound rendering in scene description languages has been a slow process. This evolutionary process is about to culminate in version 2 of MPEG-4. Version 2 of MPEG-4 contains proposals for a sound spatialization paradigm called ‘Environmental Spatialization of Audio’ (ESA) that incorporates the two sound models described earlier and all of the four stages of the auditory process. At a global level, ESA can be divided into a Physical Model and a Perceptual Model.

4.2.1 Physical Model

The physical model enables the rendering of source directivity, detailed room acoustics and acoustic properties for geometrical objects (walls, furniture, etc.). ‘Auralization’, another term for the physical model, has been defined as:

> “creating a virtual auditory environment that models an existent or non-existent space. The relation of auralization (or environmental spatialization) to graphics (visualization) is understood as the creation of audiovisual scenes that are perceptually (visually and aurally) relevant.” [8]

Three new Nodes have been devised to facilitate the physical approach. These are **AcousticScene**, **AcousticMaterial** and **DirectiveSound** (see Appendix C).

Briefly, **DirectiveSound** is a replacement for the simpler **Sound** Node. It defines a directional sound source whose attenuation can be described in terms of distance and air absorption. The direction of the source is not limited to a directional vector or a particular geometrical shape. Two fields, ‘direction’ and ‘directivity’ define the radiation pattern of the source. The ‘direction’ field specifies the direction of the sound source in terms of a 3D coordinate, this in-turn becomes the angle 0 degrees in the ‘directivity’ field. The ‘directivity’ field specifics the frequency-dependent gain as a function of azimuth around the source.

The velocity of the sound can be controlled via the ‘speedOfSound’ field; this can be used, for example, to refine the use of Doppler Effect. Attenuation over the ‘distance’ field can now drop to –60dB and can be frequency-dependent if the ‘useAirabs’ field is set to TRUE.

The ‘spatialize’ field behaves the same as its counterpart in the **Sound** Node but with the addition that any reflections associated with this source are also spatially rendered. The ‘roomEffect’ field controls the enabling of ESA and if TRUE the source is spatialized according to the environment’s acoustic parameters.

AcousticScene is a node for generating the acoustic properties of an environment. It simply establishes the volume and size of the rectangular environment and assigns it a reverberation time. The auralization of the environment involves the processing of information from the **AcousticScene** and the acoustic properties of surfaces as declared in **AcousticMaterial**.

The fields that determine the acoustic properties of surfaces in **AcousticMaterial** are 'reffunc' and 'transfunc'. The reflection characteristics of the material's surface are declared in the 'reffunc' field. Basically, this is a reflectivity transfer function that enables frequency-independent attenuation of reflections from a surface. If the value of this field is 0 then there is no reflectivity and if it is set to 1 the amplitude of the reflections will be the same as that of the incident sound. The 'transfunc' field determines the amount of energy that is allowed pass through the material, these are also known as the transmission properties of the material.

4.2.2 Perceptual Model

Version 1 of the MPEG-4 standard treated spatial sound from a physical point-of-view. Whilst this is desirable in a virtual environment it is quite limited. Virtual worlds are not constrained by physical laws and properties; therefore it was necessary to introduce a perceptual equivalence of the physical model. To this end, it is proposed that in version 2 of MPEG-4 two new Nodes should be added; **PerceptualScene** and **PerceptualSound** (see Appendix C).

Rault et al, point out the merits of the perceptual approach in a recent document to the MPEG group

> "A first advantage we see in this concept is that both the design and the control of MPEG4 Scenes is more intuitive compared to the physical approach, and manipulating these parameters does not require any particular skills in Acoustics. A second advantage is that one can easily attribute individual acoustical properties for each sound present in a given virtual scene." [9]

The principles of the perceptual model are drawn from research carried out on the Spatialisateur project, and additional elements are derived from Creative Lab's Environmental Audio Extensions (EAX) and Microsoft's DirectSound API [10].
Using the perceptual model, each sound source's spatial attributes can be manipulated individually, or an acoustic-preset can be designed for the environment (only *relative* source positions and orientations are considered in this model).

Fields such as 'Presence', 'Brilliance', and 'Heavyness' are used to configure the room/object's acoustic characteristics. In all, there are nine fields used to describe, in non-technical terms, the spatial characteristics of a room or a sound object. These fields have been derived from psycho-acoustic experiments carried out at IRCAM (Spatialisateur Project). The experiments consisted of listening tests where listeners were asked "to quantify the perceptual dissimilarity of sound fields reconstructed artificially in an anechoic room with frontal direct sound" [9]. Of the nine subjective fields, six describe perceptual attributes of the environment, and three are perceived characteristics of the source. Table 2 lists the parameters for both Environment and Source.

Table 2

Environment Fields	Source Fields
LateReverberance	Presence
Heavyness	Warmth
Liveness	Brilliance
RoomPresence	
RunningReverberance	
RoomEnvelopment	

It can also be seen from Table 2 that the last three fields of the Environment section and all of the Source fields are dependent upon the position, orientation and directivity of the source.

The validity of this approach could be questioned in terms of its subjectivity, for example, the choice of words such as ‘Warmth’ and ‘Brilliance’. However, the use of subjective terms as acoustic parameters, in this context, is to facilitate the non-specialist to compose a soundscape with convincing acoustic properties.

In terms of the Audio Framework, version 2 of MPEG-4 seems to cover all of the four stages using ESA. This facilitates physical modeling, acoustic-presets for environments and the rendering of absolute sound sources. Sound presentation is successfully addressed in both the physical and perceptual paradigms.

5 Conclusions

Spatial presentation of sound is a very important feature of Virtual Reality. Without it the virtual environment would lack the immersive qualities required for a convincing virtual experience. We have seen how the spatialization of sound has taken a relatively long time to evolve in comparison to the visual senses. While it is easy to criticize VRML for its primitive support for sound presentation it must be remembered that its origins spring from 3D Graphic Design. VRML should really be considered as a learning experience in sound spatialization for Virtual Environments.

MPEG-4 version 2 is where the main advancements in the presentation of sound have been achieved. To cater for both ESA and absolute sound rendering a dual approach has been developed. The two models, physical and perceptual, seem to encapsulate all of the necessary attributes for a cogent spatial experience. Between them, all four stages of the Auditory Process are addressed.

It now remains for developers to create great tools that will enable users to generate sound-worlds where the user feels truly enveloped in sound and immersed in the overall experience.

6 References

[1] ISO/IEC JTC/SC24 IS 14772-1 “The Virtual Reality Modeling Language (VRML97) Information technology – Computer graphics and image processing – The Virtual Reality Modeling Language (VRML)”.
URL: http://www.vrml.org/Specifications/VRML97/.April1997

[2] ISO/IEC FDIS 14496-3, 1999.

[3] W.G. Gardner, “3D Audio and Acoustic Environment Modeling”, Wave Arts Inc., March 1999. URL: http://www.wavearts.com

[4] J-M. Jot and J-B. Rault, ISO/IEC JTC1/SC29/WG11 N2578 “Extensions of Advanced AudioBIFS: a Perceptual Paradigm for the Environmental Spatialization of Audio”.

[5] N. Thalmann and D. Thalmann, “Artificial Life and Virtual Reality”, John Wiley & Son, W. Sussex, London, 1994.

[6] ISO/IEC 10646-1:1993 “Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane, Internet standards track protocol”.
URL: http://www.iso.ch/isob/switch-engine-cate.pl?searchtype=refnumber&KEYWORDS=10646

[7] J. Herder, “Tools And Widgets For Spatial Sound Authoring”, Computer Graphics, 1997.
URL: http://www.u-aizu.ac.jp/~herder

[8] R. Väänänen, “Verification Model of Advanced BIFS (Systems VM 4.0 subpart 2)”, ISO/IEC JTC1/SC29/WG11 N2525

[9] J-B. Rault et al., “Audio Rendering of Virtual Room Acoustics and Perceptual Description of the Auditory Scene”, ISO/IEC JTC1/SC29/WG11 M4222

[10] J-M. Jot, L. Ray and L. Dahl, “Extensions of Audio BIFS: Interfaces and Models Integrating Geometrical and Perceptual Paradigms for the Environmental Spatialization of Audio”, ISO/IEC JTC1/SC29/WG11 M4223

[11] ISO/IEC FDIS 14496-3 sub5, Structured Audio, 1999.

[12] J. Signès, “Binary Format for Scene (BIFS): Combining MPEG-4 media to build rich multimedia services”. URL: http://www.cselt.stet.it/mpeg/documents/koenen/signes.zip

[13] E. Scheirer et al., “AudioBIFS: The MPEG-4 Standard for Effects Processing” COST-G6 Workshop on Digital Audio Effects Processing (DAFX’98), Barcelona, Nov. 1998

[14] R. Väänänen and J. Huopaniemi, ‘Spatial Presentation of Sound in Scene Description Languages’, 17th AES International Conference, Munich, May 1999.

[15] R. Väänänen and J. Huopaniemi, “Update of advanced Audio BIFS: The Physical Approach”, ISO/IEC JTC1/SC29/WG11 M4590.

[16] J-M. Jot and J-B. Rault, ISO/IEC JTC1/SC29/WG11 M4449 “Extensions of Advanced AudioBIFS: a Perceptual Paradigm for the Environmental Spatialization of Audio”.

[17] J-M. Jot, “Synthesizing Three-Dimensional Sound Scenes in Audio or Multimedia Production and Interactive Human-Computer Interfaces”, 5th International Conference: Interface to Real & Virtual Worlds, May 1996.
URL: http://mediatheque.ircam.fr/articles/textes/Jot96a

[18] J-M. Jot, “Efficient models for reverberation and distance rendering in computer music and virtual audio reality”, Proc. 1997 Int. Computer Music Conference, pp. 236-243, September 1997. URL: http://mediatheque.ircam.fr/articles/textes/Jot97b/

Appendix A

VRML Sound Node

Sound	{		
exposedField	SFVec3f	direction	0, 0, 1
exposedField	SFFloat	intensity	1
exposedField	SFVec3f	location	0, 0, 0
exposedField	SFFloat	maxBack	10
exposedField	SFFloat	maxFront	10
exposedField	SFFloat	minBack	1
exposedField	SFFloat	minFront	1
exposedField	SFFloat	priority	0
exposedField	SFNode	source	NULL
Field	SFBool	spatialize	TRUE
}			

Appendix B

MPEG-4 version 1 Sound Node and AudioBIFS

Sound	{		
exposedField	SFVec3f	direction	0, 0, 1
exposedField	SFFloat	intensity	1
exposedField	SFVec3f	location	0, 0, 0
exposedField	SFFloat	maxBack	10
exposedField	SFFloat	maxFront	10
exposedField	SFFloat	minBack	1
exposedField	SFFloat	minFront	1
exposedField	SFFloat	priority	0
exposedField	SFNode	source	NULL
Field	SFBool	spatialize	TRUE
}			

AudioSource	{		
exposedField	MFNode	children	NULL
exposedField	MFString	url	NULL
exposedField	SFFloat	pitch	1
exposedField	SFFloat	speed	1
exposedField	SFTime	startTime	0
exposedField	SFTime	stopTime	0
field	SFInt32	numChan	1
field	MFInt32	phaseGroup	NULL
}			

AudioMix	{		
exposedField	MFNode	children	NULL
exposedField	SFInt32	numInputs	1
exposedField	MFFloat	matrix	NULL
field	SFInt32	numChan	1
field	MFInt32	phaseGroup	NULL
}			

AudioSwitch	{		
exposedField	MFNode	children	NULL
exposedField	MFInt32	whichChoice	NULL
field	SFInt32	numChan	1
field	MFInt32	phaseGroup	NULL
}			

AudioDelay	{		
exposedField	MFNode	children	NULL
exposedField	SFTime	delay	0
field	MFInt32	numChan	1
field	MFInt32	phaseGroup	NULL
}			

AudioFX	{		
exposedField	MFNode	children	NULL
exposedField	SFString	orch	" "
exposedField	SFString	score	" "
exposedField	MFFloat	params	NULL
field	SFInt32	numChan	1
field	MFInt32	phaseGroup	NULL
}			

ListeningPoint	{		
eventIn	SFBool	set_bind	NULL
exposedField	SFBool	jump	TRUE
exposedField	SFRotation	orientation	0, 0, 1, 0
exposedField	SFVec3f	position	0, 0, 10
field	SFString	description	" "
eventOut	SFTime	bindTime	
eventOut	SFBool	isBound	
}			

Appendix C

MPEG-4 version 2 Advanced Audio Nodes

Physical AcousticScene	{		
exposedField	SFFloat	paramfs	0
field	SFVec3f	3DVolumeCenter	0, 0, 0
field	SFVec3f	3DVolumeSize	-1, -1, -1
exposedField	MFFloat	reverbtime	0
}			

AcousticMaterial	{		
exposedField	SFFloat	reffunc	0
exposedField	SFFloat	transfunc	1
exposedField	SFFloat	ambientIntensity	0.2
exposedField	SFColor	diffuseColor	0.8, 0.8, 0.8
exposedField	SFColor	emissiveColor	0, 0, 0
exposedField	SFFloat	shininess	0.2
exposedField	SFColor	specularColor	0, 0, 0
exposedField	SFFloat	transparency	0
}			

DirectiveSound	{		
exposedField	SFVec3f	direction	0, 0, 1
exposedField	SFFloat	intensity	1
field	MFFloat	directivity	1
exposedField	SFFloat	speedOfSound	340
exposedField	SFFloat	distance	100
exposedField	SFVec3f	location	0, 0, 0
exposedField	SFNode	source	NULL
exposedField	MFBool	useAirabs	FALSE
exposedField	SFBool	spatialize	TRUE
exposedField	SFBool	roomEffect	TRUE
}			

Perceptual

PerceptualScene {

eventIn	MFNode	AddChildren	NULL
eventIn	MFNode	RemoveChildren	NULL
exposedField	MFNode	Children	NULL
Field	SFVec3f	BboxCenter	0, 0, 0
Field	SFVec3f	BboxSize	-1, –1, –1
exposedField	MFBool	UseAirabs	FALSE
exposedField	MFBool	UseAttenuation	TRUE
exposedField	SFFloat	RefDistance	1
exposedField	SFFloat	Latereverberance	TBD
exposedField	SFFloat	Heavyness	TBD
exposedField	SFFloat	Liveness	TBD
exposedField	MFFloat	RoomPresence	TBD
exposedField	MFFloat	RunningReverberance	TBD
exposedField	MFFloat	RoomEnvelopment	TBD
exposedField	SFFloat	Presence	TBD
exposedField	SFFloat	Warmth	TBD
exposedField	SFFloat	Brillance	TBD
exposedField	SFFloat	Fmin	250
exposedField	SFFloat	Fmax	4000

}

PerceptualSound {

exposedField	SFVec3f	direction	0.0, 0.0, 1.0
exposedField	SFFloat	intensity	1.0
exposedField	MFFloat	directivity	1.0
exposedField	MFFloat	omniDirectivity	1.0
exposedField	SFFloat	speedOfSound	340.0
exposedField	SFFloat	distance	1000.0
exposedField	SFVec3f	location	0, 0, 0
exposedField	MFFloat	relPParams	1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0
exposedField	MFFloat	directFilter	1.0, 1.0, 1.0
exposedField	MFFloat	inputFilter	1.0, 1.0, 1.0
exposedField	MFBool	useAirabs	FALSE
exposedField	MFBool	useAttenuation	TRUE
exposedField	SFInt	spatialize	FALSE
exposedField	SFInt	roomEffect	FALSE
exposedField	SFNode	source	NULL

}